Experimentally Verified Design and Simulation of Fluxonium Systems

PHYS559 Final Project

Sadman Ahmed Shanto[1] Anshu[1] Amy F. Brown[1]

[1]Department of Physics
University of Southern California

December 11, 2023

Table of Contents

Process Summary

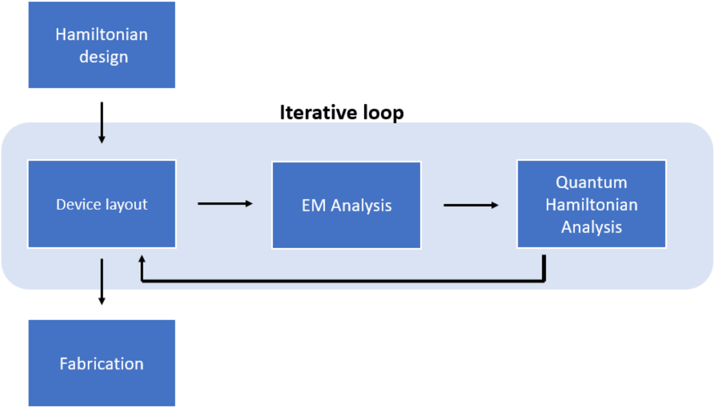


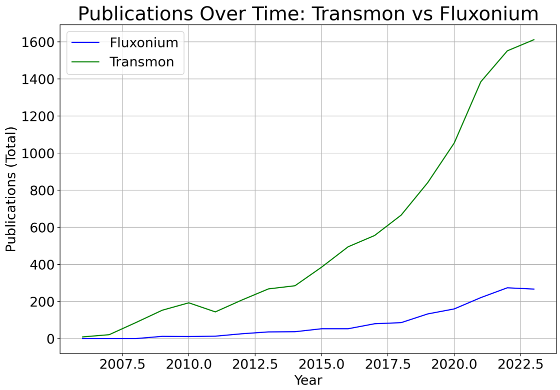
Figure: Summary of SCQ Design Process

Bottlenecks

- Designing physical quantum circuits is a non-trivial problem and the iterative process creates a bottleneck
- Reality is used as simulation because its faster in a lot of cases
- This is an area of active multi-faceted research - scqubits, sqcircuits, open quantum tools, Machine learning in EDA
- Simulator(hyperparameters, validation data) $\approx$ reality
- Simulation pipeline $\rightarrow$ SQuADDs (Superconducting Qubit And Device Design and Simulation database)

Our Problem

- Fluxonium qubit is a promising candidate - we want to create easy access to doing science with it



Our Problem

- Fluxonium qubit is a promising candidate - we want to create easy access to doing science with it

- Create an open-source tool to design fluxonium circuits
- Create a validated simulation framework for fluxonium similar to Transmons
- Contribute the data to SQuADDS to help others with their design-to-fab process

Table of Contents

Quick Review: Classical LC Resonator Circuit

- From node voltages $V(t)$ and currents $I(t)$ to instantaneous energies $E(t)$ and generalized coordinates $Q(t), \Phi(t)$:

$$E(t) = \int_{-\infty}^{t} V(t')I(t')dt'$$

$$Q(t) = \int_{-\infty}^{t} I(t')dt'$$

$$\Phi(t) = \int_{-\infty}^{t} V(t')dt'$$

Quick Review: Classical LC Resonator Circuit

- From node voltages $V(t)$ and currents $I(t)$ to instantaneous energies $E(t)$ and generalized coordinates $Q(t), \Phi(t)$:

$$E(t) = \int_{-\infty}^{t} V(t')I(t')dt'$$

$$Q(t) = \int_{-\infty}^{t} I(t')dt'$$

$$\Phi(t) = \int_{-\infty}^{t} V(t')dt'$$

- leads to classical LC circuit Hamiltonian:

$$\begin{aligned} H &= \dot{\Phi} p_{\Phi} - L \\ &= \frac{1}{2} C \dot{\Phi}^2 + \frac{1}{2L} \Phi^2 \\ &= \frac{1}{2C} Q^2 + \frac{1}{2L} \Phi^2 \end{aligned}$$

Quick Review: Quantum Harmonic Oscillator(QHO)

- Quantize charge and flux operators

$$
\begin{aligned}
H &= \frac{1}{2C} Q^2 + \frac{1}{2L} \Phi^2 \\
&= 4 \left(\frac{e^2}{2C} \right) \left(\frac{Q}{2e} \right)^2 + \frac{1}{2} \left(\frac{\Phi_0}{2\pi} \right)^2 \frac{1}{L} \left(\frac{2\pi\Phi}{\Phi_0} \right)^2 \\
&= 4E_C n^2 + \frac{1}{2} E_L \phi^2 \\
&= 4E_C \left[\left(\frac{E_L}{32E_C} \right)^{1/4} i \left(a - a^\dagger \right) \right]^2 + \frac{1}{2} E_L \left[\left(\frac{2E_C}{E_L} \right)^{1/4} \left(a + a^\dagger \right) \right]^2 \\
&= \sqrt{8E_L E_C} \left(a^\dagger a + \frac{1}{2} \right) \\
&= \hbar\omega \left(a^\dagger a + \frac{1}{2} \right), \qquad \omega = \sqrt{8E_L E_C}/\hbar
\end{aligned}
$$

Quick Review: Fixed-Frequency Transmon

- Replace linear inductor with Josephson junction, which obeys Josephson relations:

$$I = I_c \sin(\phi), \qquad V = \frac{\hbar}{2e}\frac{d\phi}{dt},$$

Quick Review: Fixed-Frequency Transmon

- Replace linear inductor with Josephson junction, which obeys Josephson relations:

$$I = I_c \sin(\phi), \qquad V = \frac{\hbar}{2e}\frac{d\phi}{dt},$$

- Calculate new potential energy

$$\begin{aligned} U_J &= \int VIdt \\ &= \int \frac{\hbar}{2e}\frac{d\phi}{dt} I_c \sin(\phi) dt \\ &= -E_J \cos(\phi) \end{aligned}$$

Quick Review: Fixed-Frequency Transmon

- Replace linear inductor with Josephson junction, which obeys Josephson relations:

$$I = I_c \sin(\phi), \qquad V = \frac{\hbar}{2e}\frac{d\phi}{dt},$$

- Calculate new potential energy

$$\begin{aligned} U_J &= \int VIdt \\ &= \int \frac{\hbar}{2e}\frac{d\phi}{dt} I_c \sin(\phi) dt \\ &= -E_J \cos(\phi) \end{aligned}$$

- Fixed-frequency transmon Hamiltonian:

$$H = 4E_C n^2 - E_J \cos(\phi)$$

Quick Review: Fixed-Frequency Transmon

- Fixed-frequency transmon Hamiltonian:

$$
\begin{aligned}
H &= 4E_C n^2 - E_J \cos(\phi) \\
&= 4E_C n^2 + \frac{1}{2}E_J\phi^2 - \frac{1}{24}E_J\phi^4 + \mathcal{O}(\phi^6) \\
&= 4E_C \left[\left(\frac{E_J}{32E_C}\right)^{1/4} i\left(a - a^\dagger\right)\right]^2 + \frac{1}{2}E_J\left[\left(\frac{2E_C}{E_J}\right)^{1/4}\left(a + a^\dagger\right)\right]^2 \\
&\quad - \frac{1}{24}E_J\left[\left(\frac{2E_C}{E_J}\right)^{1/4}\left(a + a^\dagger\right)\right]^4 + \mathcal{O}(\phi^6)
\end{aligned}
$$

- Fixed-frequency transmon Hamiltonian:

$$\begin{aligned} H &= 4E_C n^2 - E_J \cos(\phi) \\ &= 4E_C n^2 + \frac{1}{2}E_J\phi^2 - \frac{1}{24}E_J\phi^4 + \mathcal{O}(\phi^6) \\ &= 4E_C \left[\left(\frac{E_J}{32E_C}\right)^{1/4} i\left(a - a^\dagger\right)\right]^2 + \frac{1}{2}E_J\left[\left(\frac{2E_C}{E_J}\right)^{1/4}\left(a + a^\dagger\right)\right]^2 \\ &\quad - \frac{1}{24}E_J\left[\left(\frac{2E_C}{E_J}\right)^{1/4}\left(a + a^\dagger\right)\right]^4 + \mathcal{O}(\phi^6) \end{aligned}$$

- Retaining only Fock-number-preserving terms, we have

$$\begin{aligned} H &= \left(\sqrt{8E_JE_C} - E_C\right) a^\dagger a - \frac{1}{2}E_C a^\dagger a^\dagger aa \\ &= \omega_q a^\dagger a + \frac{1}{2}\alpha a^\dagger a^\dagger aa, \qquad \omega_q = \sqrt{8E_JE_C} - E_C, \alpha = -E_C \end{aligned}$$

Quick Review: Fluxonium

- Wire branch in parallel with JJ; increase number of junctions on one arm to ~100 [1], we arrive at the fluxonium Hamiltonian:

$$H = 4E_C n^2 - E_J \cos(\phi + \varphi_e) + \frac{1}{2} E_L \phi^2.$$

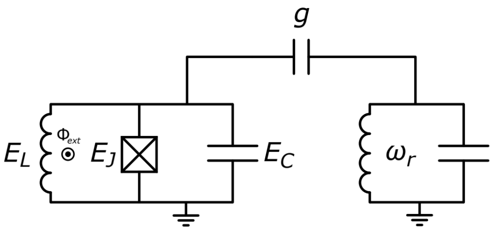


Figure: Fluxonium circuit

Simulation outputs to Hamiltonian Parameters

SQuADDS [2] treatment of fixed-frequency transmon

$$H_q = 4E_{C,q}n_q^2 - E_J\cos(\phi_q)$$

capacitively coupled to a QHO resonator

$$H_r = 4E_{C,r}n_r^2 + \frac{1}{2}E_L\phi_r^2$$

with interaction Hamiltonian

$$H_{int} = 4e^2\frac{C_c}{C_qC_r}n_qn_r$$

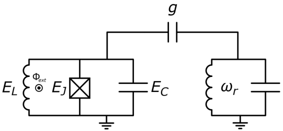


Figure: Fluxonium circuit

Simulation outputs to Hamiltonian Parameters

- In the SQuADDS paper [2], they determined that

$$g \approx \frac{C_c}{C_q}\sqrt{\frac{e^2\omega_r}{C_r}}\left(\frac{E_j}{8E_{C,q}}\right)^{1/4}$$

Table of Contents

Fluxonium Device Design from Professor Pechenezhskiy

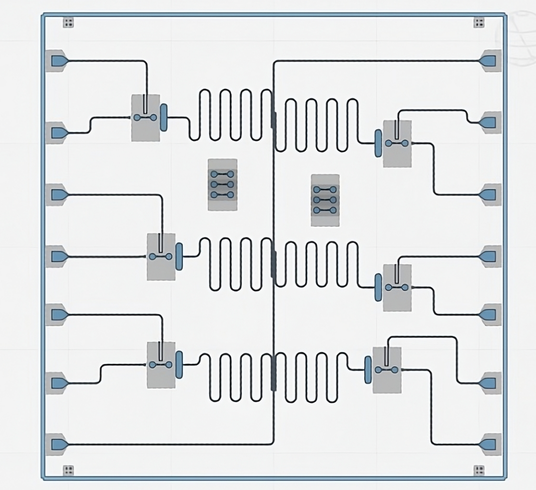
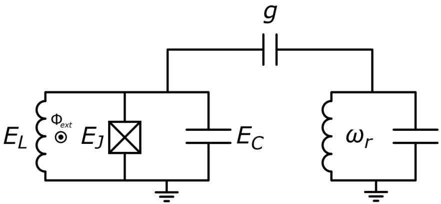
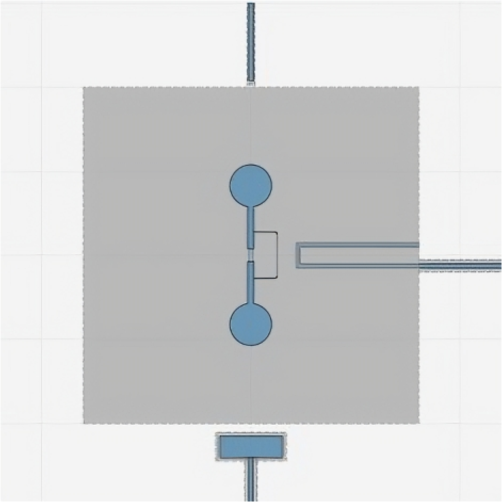


Figure: Zucchini recreated in Qiskit Metal

Design Tool Creation





Design Tool Creation

- Parametric creation of the Zucchini Fluxonium Qubit in Qiskit Metal

Design Tool Creation

- Parametric creation of the Zucchini Fluxonium Qubit in Qiskit Metal
- Refactored AnsysHFSS Renderer and Custom QGmsh Renderer

Design Tool Creation

- Parametric creation of the Zucchini Fluxonium Qubit in Qiskit Metal
- Refactored AnsysHFSS Renderer and Custom QGmsh Renderer
- "Plug and play" compatibility with LOM and EPR Analysis from Qiskit Metal API

Connecting to Ansys

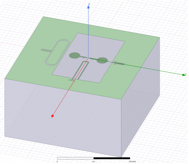


Figure: Cap. and Ind. Mat. Extraction from Fluxonium

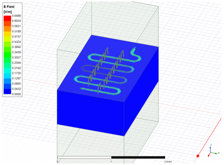


Figure: E field distribution of cavity

- Rendering was trivial but TEDIOUS
- Runs native to Transmons in qiskit-metal API

Connecting to Ansys

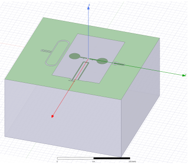


Figure: Cap. and Ind. Mat. Extraction from Fluxonium

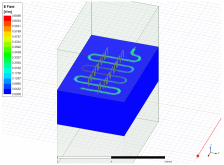


Figure: E field distribution of cavity

- Rendering was trivial but TEDIOUS
- Runs native to Transmons in qiskit-metal API
- Refactored EPR code to handle non-linear terms non-perturbatively [https://arxiv.org/pdf/2309.17286v1.pdf]

Connecting to Ansys

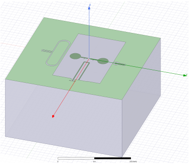


Figure: Cap. and Ind. Mat. Extraction from Fluxonium

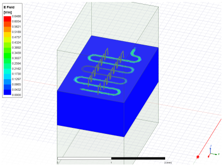


Figure: E field distribution of cavity

- Rendering was trivial but TEDIOUS
- Runs native to Transmons in qiskit-metal API
- Refactored EPR code to handle non-linear terms non-perturbatively [https://arxiv.org/pdf/2309.17286v1.pdf]
- Flow:
  - Design $\rightarrow$ Ansys $\rightarrow$ Simulation Outputs (e.g. cap. matrix, ind. matrix, eigenmode data, etc) $\rightarrow$ Qiskit Metal Analyses (e.g. LOM, EPR, etc)

Connecting to Palace

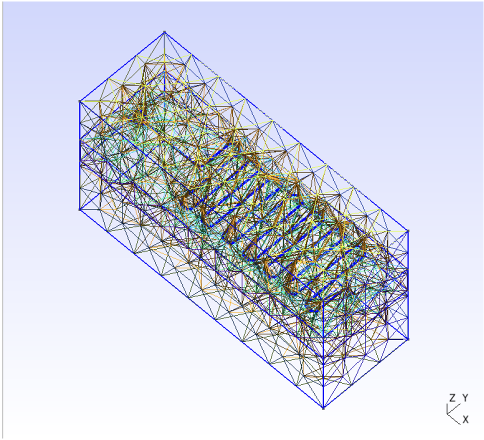


Figure: CPW Mesh Input to Palace

- Palace requires a mesh file and configuration file as an input
- Mesh file defines the geometry and configuration file defines the materials and physics
- Hyper-parameter turning needed similar to Ansys

Connecting to Palace

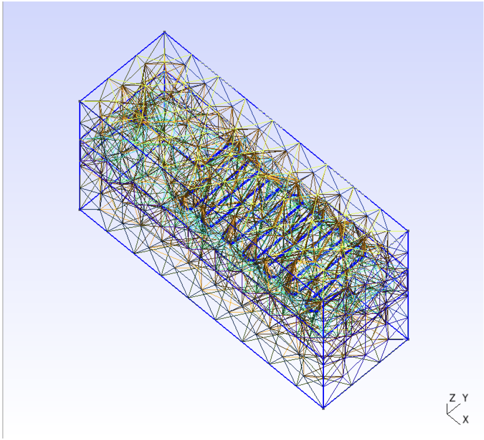


Figure: CPW Mesh Input to Palace

- Palace requires a mesh file and configuration file as an input
- Mesh file defines the geometry and configuration file defines the materials and physics
- Hyper-parameter turning needed similar to Ansys
- Flow:
  - Design → Mesh

Connecting to Palace

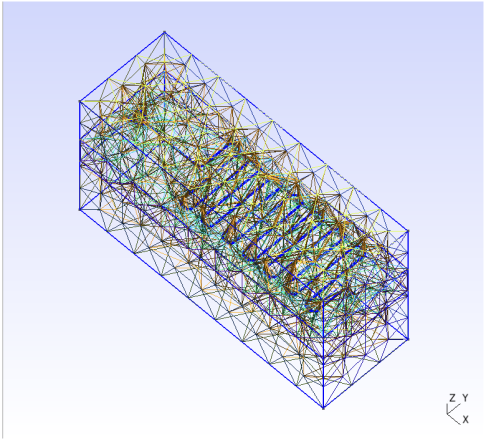


Figure: CPW Mesh Input to Palace

- Palace requires a mesh file and configuration file as an input
- Mesh file defines the geometry and configuration file defines the materials and physics
- Hyper-parameter turning needed similar to Ansys
- Flow:
  - (Mesh + Config) $\rightarrow$ Palace

Connecting to Palace

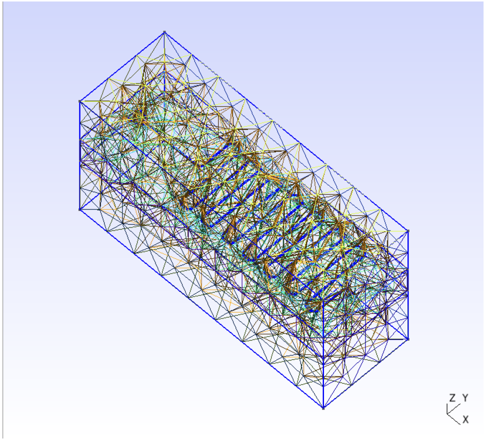


Figure: CPW Mesh Input to Palace

- Palace requires a mesh file and configuration file as an input
- Mesh file defines the geometry and configuration file defines the materials and physics
- Hyper-parameter turning needed similar to Ansys
- Flow:
    - Palace $\rightarrow$ Simulation Output Files (e.g. E field distr., cap. matrix, eigenmode data, etc)

Connecting to Palace

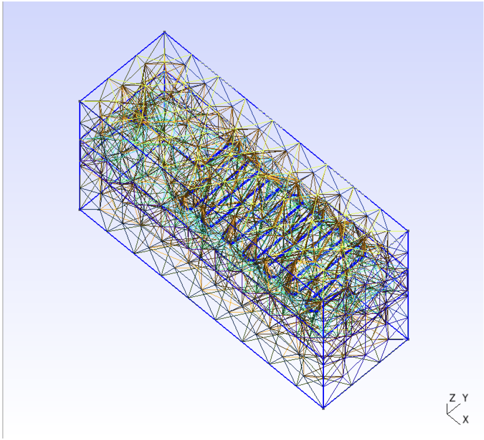


Figure: CPW Mesh Input to Palace

- Palace requires a mesh file and configuration file as an input
- Mesh file defines the geometry and configuration file defines the materials and physics
- Hyper-parameter turning needed similar to Ansys
- Flow:
  - Simulation Output Files $\rightarrow$ Your Analysis Code

Simulation Methodology

- Ran the pyEPR simulation and analysis on our simulation unit - *fluxonium* + *cpw* + *coupler* + *transmission line stub*

Simulation Methodology

- Ran the pyEPR simulation and analysis on our simulation unit - *fluxonium* + *cpw* + *coupler* + *transmission line stub*
- Ran a hybrid LOM Analysis in Ansys HFSS
  - capacitance matrix and inductance matrix extraction of *fluxonium* only
  - eigenmodal simulation of *cpw* + *coupler* + *transmission line stub*

Simulation Methodology

- Ran the pyEPR simulation and analysis on our simulation unit - *fluxonium* + *cpw* + *coupler* + *transmission line stub*
- Ran a hybrid LOM Analysis in Ansys HFSS
  - capacitance matrix and inductance matrix extraction of *fluxonium* only
  - eigenmodal simulation of *cpw* + *coupler* + *transmission line stub*
- Ran the same LOM Analysis above but with AWS Palace

Simulation Methodology

- Ran the pyEPR simulation and analysis on our simulation unit - *fluxonium* + *cpw* + *coupler* + *transmission line stub*
- Ran a hybrid LOM Analysis in Ansys HFSS
  - capacitance matrix and inductance matrix extraction of *fluxonium* only
  - eigenmodal simulation of *cpw* + *coupler* + *transmission line stub*
- Ran the same LOM Analysis above but with AWS Palace

Results: (Design Parameters $\mapsto$ Hamiltonian Parameters) s.t. compatible with SQuADDS

Simulation Results

Comparison of Measured and Simulated Qubit Parameters.

	Measured	HFSS (LOM)	HFSS (EPR)	Palace
E_J (GHz)	1.29	1.29	1.29	1.29
E_L (GHz)	0.87	0.54	...	0.87*
E_C (GHz)	1.31	1.26	...	1.29
$f_q(\phi = 0)$ (GHz)	2.64	3.90	7.12*	2.61
f_r (GHz)	5.55	5.78	5.48	5.73
g (MHz)	40	TBD	81*	TBD

Values in blue are user input parameters and values in orange are computed using scqubits

Understanding the Results: HFSS (LOM)

Table: Comparison of Measured and HFSS (LOM) Qubit Parameters

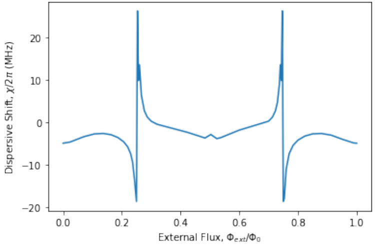
	Measured	HFSS (LOM)
E_J (GHz)	1.29	1.29
E_L (GHz)	0.87	0.54
E_C (GHz)	1.31	1.26
$f_q(\phi = 0)$ (GHz)	2.64	3.90
f_r (GHz)	5.55	5.78
g (MHz)	40	TBD

- Variation in E_L from measured value $\Longrightarrow$ issues with I mat. extraction
- Resonator frequency f_r and cap. mat. (E_C) are closer to measured values.
- f_q error as a consequence of E_L

Understanding the Results: HFSS (EPR)

Table: Comparison of Measured and HFSS (EPR) Qubit Parameters

	Measured	HFSS (EPR)
E_J (GHz)	1.29	1.29
E_L (GHz)	0.87	...
E_C (GHz)	1.31	...
$f_q(\phi = 0)$ (GHz)	2.64	7.12*
f_r (GHz)	5.55	5.48
g (MHz)	40	81*



- Bug in inductive loop, L, value....

Understanding the Results: Palace

Table: Comparison of Measured and Palace Qubit Parameters

	Measured	Palace
E_J (GHz)	1.29	1.29
E_L (GHz)	0.87	0.87*
E_C (GHz)	1.31	1.29
$f_q(\phi = 0)$ (GHz)	2.64	2.61
f_r (GHz)	5.55	5.73
g (MHz)	40	TBD

- Could not get extract L matrix from palace because of incorrectly defined config
  - issue is in defining currents through JJ *(most likely)*
  - got 0.276 GHz once, but not repeatable
- Eigenmodal and capacitance matrix results are reliably accurate

Table of Contents

Summary

- Motivation: Democratize and accelerate research of fluxonium systems by reducing the barriers to entry

Summary

- Motivation: Democratize and accelerate research of fluxonium systems by reducing the barriers to entry
- Deliverables:
  - open-source code in qiskit-metal to parametrically generate fluxonium designs
  - simulate and analyze fluxonium systems in Ansys HFSS
  - simulate and analyze fluxonium systems in AWS Palace

Summary

- Motivation: Democratize and accelerate research of fluxonium systems by reducing the barriers to entry
- Deliverables:
  - open-source code in qiskit-metal to parametrically generate fluxonium designs
  - simulate and analyze fluxonium systems in Ansys HFSS
  - simulate and analyze fluxonium systems in AWS Palace
- Future Work:
  - Finish deriving g for the fluxonium-cavity system
  - Address the simulation issues discovered
  - Simulate en-masse in palace and contribute to SQuADDS

Table of Contents

References

- P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver.
  A quantum engineer's guide to superconducting qubits.
  Applied Physics Reviews, 6(2), June 2019.

- Sadman Shanto, Andre Kuo, Clark Miyamoto, Haimeng Zhang, Vivek Maurya, Evangelos Vlachos, Malida Hecht, and Eli M. Levenson-Falk.
  Squadds: A validated design database and simulation workflow for superconducting qubit design.